ELSEVIER

SciVerse ScienceDirect

Physics Procedia 24 (2012) 1364 – 1368

Physics Procedia

2012 International Conference on Applied Physics and Industrial Engineering

Achieving Optimal Solution of Linear Programming Based on Mobile Agent Technology

Guo-feng Ma[1], Jun-hong Yang[1], Ping-chuan Zhang[2]

[1]*Dept. of Information Engineering*
ZHENGZHOU Railway Vocational& Technical College
ZHENGZHOU, China
[2]*Dept. of Electronic Science and Technology of HUST*
LUOHE Vocational Technology College
LUOHE, China

Abstract

Linear programming is an important solution for engineering applications, and sometimes the integer value is needed for the applications, this paper introduced the third distributed computation technology namely Mobile Agent to the linear programming problems, designed the algorithm with low time complexity compared to the traditional method, clarified the method of seeking the optimization in linear programming by means of Mobile Agent，and it's very valuable and practicable.

Keywords- Linear Programming; Mobile agent; Time complexity; Optimization

1. Introduction

Linear programming [1] is an important branch of operational research, that has been studied earlier, developed quickly, widely used method and more mature, it is a mathematical method to assist the people to carry out scientific management. In economic management, transportation, industrial and agricultural production economic activities, and it is one indispensable requirement to improve the economic effect. In general, the objective linear function of linear constrained maximum or minimum value of the issue, collectively known as the linear programming problem. The solution to meet the linear constraints is called feasible solution, by all feasible solutions of the set is called feasible region. Decision variables, constraints, objective function is the three elements of linear programming.

 doi:10.1016/j.phpro.2012.02.203

Linear programming is often an ordinary and important problem in Engineering practice, in which some of the best solution may be fractional, but for some specific issues, often required solution must be an integer. Such as solving the number of machines, the number of people completing the work or loading of vehicles, etc. the fractional result of the solution scores will not meet the requirements. It is integer solutions to meet the requirements, and at first glance, it seems that as long as the scores have been obtained with the solution through the "rounding of the whole" on it, but it often does not work, because it is not necessarily after transferred into an integer feasible solution; or it is a feasible solution, but not necessarily the optimal solution. Therefore, get the optimal integer solution to the problem and need further analysis, such problems are usually called integer linear programming problem, and it was a branch developed nearly 20 years from linear programming field.

It is usually classified into two types for Integer programming, if all integer variables are required, it is called pure integer programming or full integer programming, if only part of the variable is limited to an integer, is called mixed integer programming. Traditional method is to branch and bound algorithm which can be used to solve pure integer or mixed integer linear programming problem, it was presented in the early 60s of this century by Land Doid and D. Akin, and others. Since this method is flexible and easy to use of computer, so now it is an important solution method of integer linear programming. Given a maximum integer linear programming problem P, and its corresponding linear programming for the problem R, starting from the solutions of the problem R, if the optimal solution does not meet the conditions of integers A, then the optimal objective function R will P m * the optimal objective function upper bound, denoted mU; and any random feasible solution of A for objective function value is a lower bound m of m *. Branch and bound method is used for the feasible region R to be divided into sub-regions (called branches), and gradually decrease m^U and increase m, until get m*. But its efficiency is low; this paper discusses the evolution of Agent-based solutions for liner programming.

2. Agent-based integer linear programming program

The so-called agent [2] is a program that is able to operate and migrate between the same structure and heterogeneous networks autonomously and independently, it has autonomy, intelligence, mobility, collaboration and so on. Agent as a third-generation artificial intelligence, distributed computing technology, its theory and technology for the distribution system analysis, design and implementation provides a new way, known as "software as a major breakthrough." Agent technology learns from many other areas and integrates together, so it has been more widely used.

The defined agent [3] is a complex objective and may behave to copy, move, and communications functions, the behavior of Agent with the symbol S can be described as follows.

2.1 copy

Any Agent S can be copied into two sub-Agent Sl, S2, and S which have the same goal, but the circumstances of question have changed, said the parent S Agent, its copy body Sl, S2 are called sub-Agent, at the same time an Agent either may be a parent Agent or a sub-Agent.

2.2 Action

Agent S has always kept calculating its objectives by a certain way, until the success or reaches a stop condition.

First, set the environment E has a Agent S, it is calculated in accordance with its objectives in some way, if the target can not be achieved then it began to replicate, let S1, S2 for replicated agent. At this

point, S1, S2 have the same goal, but the environment in which to replicate when different when S, E is also divided into two sub-environments E1, E2, they were Sl, S2 in which the issue of the environment.

S1, S2 to calculate the target in their respective environments, if they can achieve the target, then send results to his father Agent S, if not, then continue to reproduce, while the parent Agent S determines whether reaches its goals according to the results of sub-agent Sl, S2 , and decide whether to interrupt S1, S2 action.

The agent model [4] described above can be used to solve the integer linear programming problem, the Slack R is seemed as the objectives and the environment of S. Firstly, the result of S may be solved in simplex method. If having achieved the S target (that is all of the optimal solution obtained are integer), then the corresponding integer programming problem has been proved. Otherwise, S replicate out Sl, S2 (corresponding to sub-problems R1, R2), so the Agent constantly continues solving, reproduce, eventually to obtain the integer solution in accordance with the steps of branch and bound, the solution algorithm[5] can be described as follows :

1) The objective function and constraints of the integer linear programming problem is assigned to Agent S as the objective and the environment of Agent, and define the variable m, m is not assigned any value at first.
2) Agent S begin to solve its goal, firstly to solve the relaxation problem, if the optimal solution to the relaxation problem are all integer, then m is the optimal value, turn to 6), if no feasible solution for the relaxation problem, transfer to 6), otherwise continue to the next step.
3) if the value of a variable yj to take fraction nj, then obtained the greatest integer [nj] which less than nj and the smallest integer [nj] +1 that greater than nj, and then start copying Agent S, set the body copy for the S1, S2, S1, S2 have the same goals to S, the constraints yj $\leqslant$ [nj] and yj> [nj] +1 were added to their environment based on that of S.
4) Agent S1, S2 calculated theirs objectives, if the optimal solution (integer) has been obtained, then
   IF m to have values
   Then m = the optimal value
   Else m = min (m, optimal value)
5) for S1 and S2, there are non-integer optimal solutions Si,
   IF no value for m or the non-integer optimal solution is less than m
   Then S = Si, transfer 3)
6) If m has value, then output m, otherwise output no feasible solution for the integer linear programming problem.

The Aglet system framework is shown in Fig. 1.

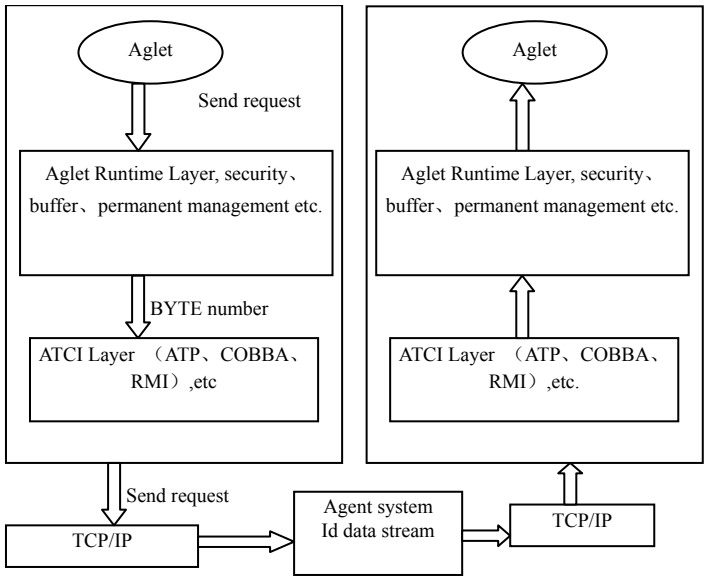

Figure 1 Aglet system framework

The Aglet is selected to achieve the above model, Aglet is a platform based on Mobile Agent technology developed by IBM's Japanese unit which is the most mature mobile agent solution currently. Aglet first create Agent S, when need to copy Agent S1, S2, Aglet implement the copy; and then assign them to their execution environment. When it's time to stop the Agent, Aglet implements the suspension or removal.

3. Algorithm's time complexity

To achieve above models, and use it to solve integer linear programming problem, the time complexity [6] is O (n) when using the general simplex method, while Agent-based models is used, its time complexity O(log n).

4. Conclusion

The linear programming problems is an important issue for engineering applications, this paper represented a new method that introduced the agent technology to it, that is the combination of linear programming problems and the Agent. A computational model was designed, and the algorithms were work out, this model achieved good results, characterized mainly reflected in: As Agent replication can be simultaneous, so the process of solving is parallel; solving process is based on goal-driven; the time complexity of the procedures is low.

References

[1] Hu Qing Huai, Yi-Ming Wei. Linear Programming and Its Applications [M]. Science Press, 2009

[2] Charlies .J. Peter. Agent-Based Engineering, the web and Intelligence [J]. IEEE Expert, December2004, pp:24-29

[3] TAO Xian-ping, etc.. Mobile component system model of analysis and description [J]. Journal of Software, 2004,12 (01) , pp:56-64

[4] Zhang Yun-yong, et al. Mobile agent technology [M]. Tsinghua University Press, 2003

[5] Woodridge, M. verifying that agents implement a communication language. In Proceedings of the 16th National Conference on Artificial Intelligence (AAAI-99), Orlando, FL, pp.52-57.

[6] YIN Zhao-lin. Mobile Intelligent Agent Technology [M].China University of Mining and Technology Press, 2006